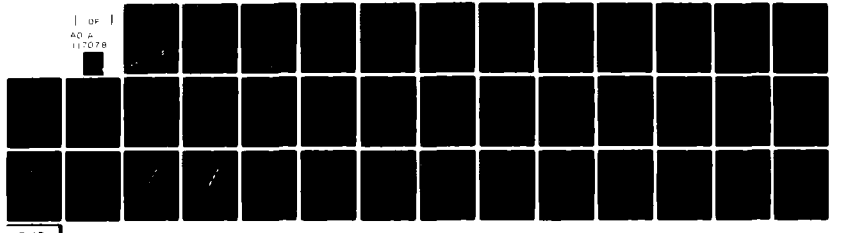
AD-A117 078 SCIENCE APPLICATIONS INC PLEASANTON CALIF F/G 12/1
MOVING FINITE ELEMENTS IN 2-D.(U)
JUN 82 R J GELINAS F49620-81-C-0073
UNCLASSIFIED SAI/PL/MFE-2D-1-82 AFOSR-TR-82-0576 NL



END
DATE FILMED
08-82
DTIC

AFOSR-TR- 82-0576

(4)

SAI/PL/MFE-2D-1-82

MOVING FINITE ELEMENTS IN 2-D

First Annual Report

Reporting Period: December 8, 1981, to June 7, 1982

AFOSR Contract: F49620-81-C-0073
Program Manager: Major C. Edward Oliver

submitted by

Robert J. Gelinas
Principal Investigator

Science Applications, Inc.
1811 Santa Rita Road
Pleasanton, CA 94566

(415) 462-5300

June 7, 1982

Approved for public release; distribution unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFOSR-TR- 82-0576	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MOVING FINITE ELEMENTS IN 2-D		5. TYPE OF REPORT & PERIOD COVERED First Annual Report December 8, 1981 to June 7, 1982
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert J. Gelinas		8. CONTRACT OR GRANT NUMBER(s) F49620-81-C-0073
9. PERFORMING ORGANIZATION NAME AND ADDRESS Science Applications, Incorporated 1811 Santa Rita Road, Suite 104 Pleasanton, California 94566		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61102F 2304/A3
11. CONTROLLING OFFICE NAME AND ADDRESS Director of Math & Informational Sciences Air Force Office of Scientific Research Attn: NM; Building 410; Bolling AFB, DC 20332		12. REPORT DATE June 7, 1982
		13. NUMBER OF PAGES 37
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Moving grid
finite element
partial differential equations
stiff ordinary differential equations
boundary conditions
Jacobian matrix
matrix solutions
conservation equations

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The moving finite element (MFE) method is a new PDE solution method which has shown significant promise in 1-D for the numerical solution of some of the most difficult problems under study with extremely large, but finite, gradients. The overall objective of the present research is to explore further the promise of the continuous node moving properties of the MFE method in 2-D. For this, both the logical structure of the MFE method and its reduction to practice in 2-D are under investigation in this project.

(continued on reverse side

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. In this first year's effort, results of several test examples involving conservation equations have shown the MFE method to be a robust PDE solver in 2-D. Burger-like equations which generate non-uniform, skewed waveforms with extremely large gradients in 2-D have been solved stably and accurately on an 8 x 8 grid of moving nodes. Examples of scalar waves propagating on closed, circular paths have been solved with similar levels of effectiveness on a 6 x 6 MFE grid mesh. The node behavior in all examples tested to date has been extremely flexible and easily controlled with no grid tangling or biasing effects. The MFE code architecture in 2-D is found to be amenable to vector and parallel processing computational methods and advanced computers.

INTRODUCTION

The moving finite element (MFE) method[1],[2] has shown significant promise in 1-D for solving numerically some of the most difficult partial differential equation (PDE) problems which may contain multiple large gradients. The objective of the present research is to investigate further the numerical properties and structure of the MFE method in order to reduce it to practice in 2-D for important PDE's which occur in the basic sciences and engineering.

I. MATHEMATICAL BACKGROUND

In order to discuss this research concisely, the basic formulation of the MFE method in 2-D is presented immediately below.

Consider a general system of PDE's, $\dot{U} = L(U)$, or

$$\begin{aligned} \dot{u}_1 &= L_1(U) \\ &\vdots \\ \dot{u}_N &= L_N(U) \quad . \end{aligned} \tag{1}$$

Using piecewise linear approximants of $u_1 \ldots u_N$, which are of the form $u = mx + ny + p$, on a hexagonally connected triangular mesh (see Figures 1-3), application of the chain rule to the differentiation of u_k gives

$$\dot{u}_k = \sum_j \dot{a}k_j \alpha k^j + \dot{x}_j \beta k^j + \dot{y}_j \gamma k^j \quad , \quad \text{where} \tag{2}$$

$$\alpha k^j = \alpha^j = \frac{\partial u_k}{\partial a k_j} \; ; \; \beta k^j = \frac{\partial u_k}{\partial x_j} \; ; \; \gamma k^j = \frac{\partial u_k}{\partial y_j} \quad . \tag{3}$$

The functions αj, βkj, and γkj are seen to be certain piecewise linear functions having their support in the hexagon of six triangles surrounding each jth node. See Figures 1-4.

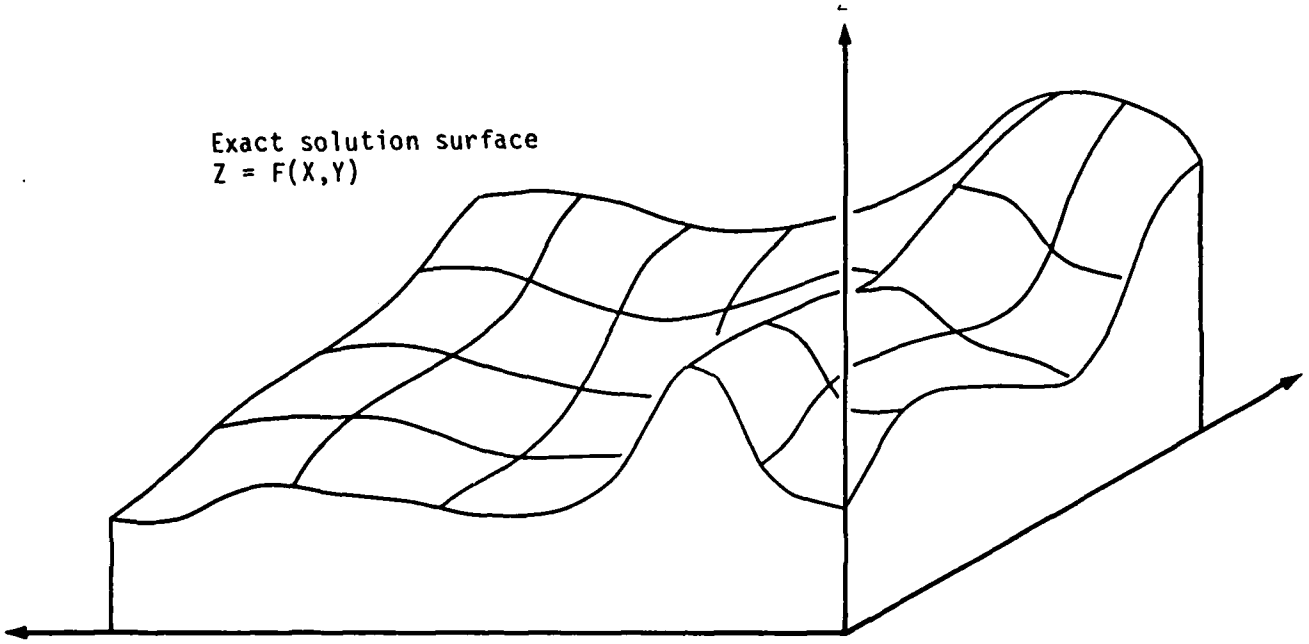


Figure 1. Exact solution surface, with lines of constant X and constant Y.

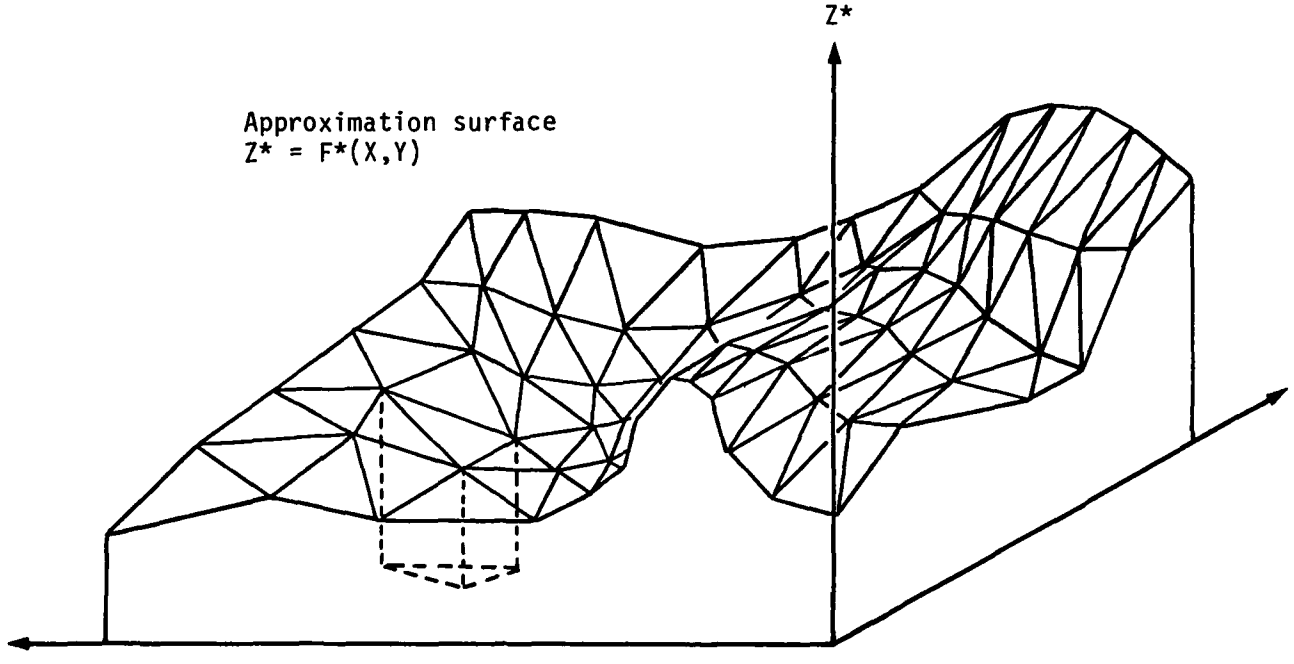


Figure 2. Approximate solution represented by piecewise linear functions making up triangular facets.

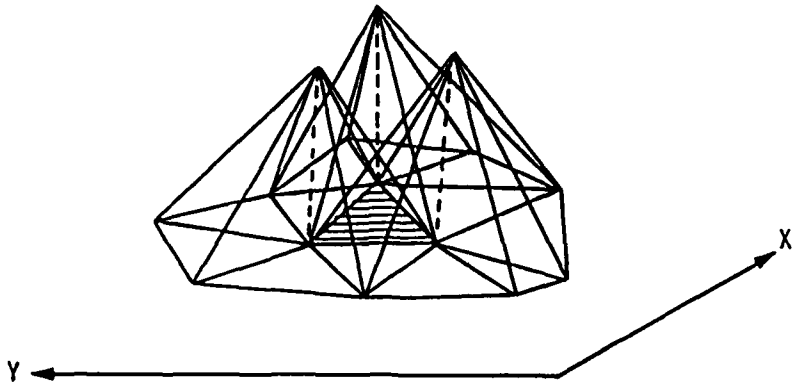


Figure 3. Basis functions defined on each hexagon provide three linearly independent basis functions on each triangle of the entire problem domain.

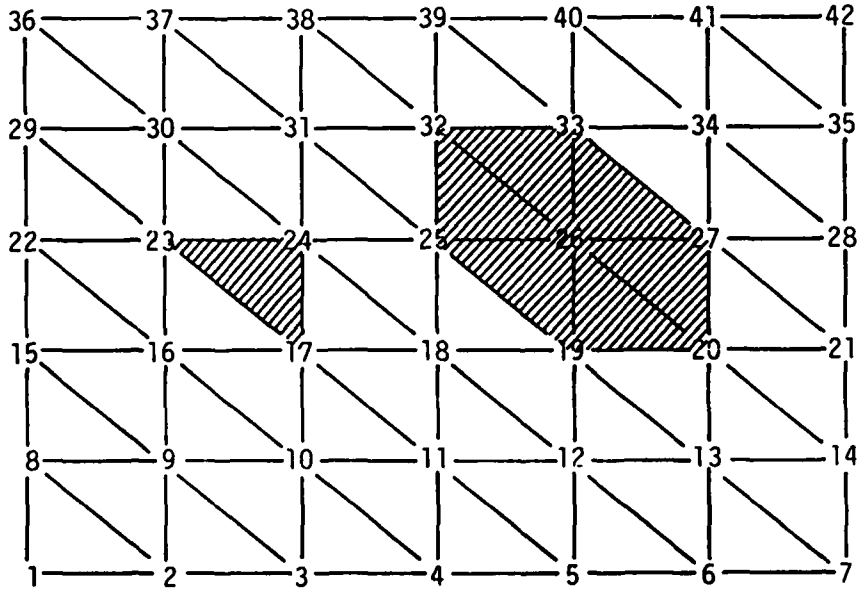


Figure 4. Nodes.

The following ODE's are derived for the parameters of the MFE method by a formal minimization of the L^2 norm of PDE residuals $[\dot{U} - L(U)]$, plus regularization terms, with respect to the parameter derivatives $\dot{a}l_j, \ldots, \dot{a}N_j, \dot{x}_j, \dot{y}_j$:

$$\sum_j \left[(\alpha^j, \alpha^i)\dot{a}k_j + (\beta k^j, \alpha^i)\dot{x}_j + (\gamma k^j, \alpha^i)\dot{y}_j \right] = (L_k(U), \alpha^i) \text{ for } k = 1, \ldots N, \tag{4a}$$

$$\sum_{k=1}^{N} \sum_j \left[(\alpha^j, \beta k^i)\dot{a}k_j + (\beta k^j, \beta k^i)\dot{x}_j + (\gamma k^j, \beta k^i)\dot{y}_j \right] = \sum_{k=1}^{N} (L_k(U), \beta k^i) + (\text{regularization terms}) \tag{4b}$$

$$\sum_{k=1}^{N} \sum_j \left[(\alpha^j, \gamma k^i)\dot{a}k_j + (\beta k^j, \gamma k^i)\dot{x}_j + (\gamma k^j, \gamma k^i)\dot{y}_j \right] = \sum_{k=1}^{N} (L_k(U), \gamma k^i) + (\text{regularization terms}). \tag{4c}$$

The sums on j in Equations (4) run over the seven neighboring nodes of i (including the i^{th} node itself) in the hexagonal grid. Equations (4) thus provide the basic working equations of the MFE method in 2-D. This system of ODE's is of the form

$$R(y) \equiv C(y)\dot{y} - g(y) = 0 \ , \tag{5}$$

where $y(t) = (al_1, \ldots, x_1, y_1; al_2, \ldots, x_2, y_2; \ldots\ldots)$ is the vector of unknown parameters, and the "mass matrix" $C(y)$ is symmetric and positive definite. This system of ODE's can be quite stiff, and stiff ODE solution methods are thus required in order to effectively obtain the numerical solutions of Equations (4) and (5).

Regularization

As was evident in the 1-D DYLA code[2,3,4,5], the MFE method is also amenable in 2-D to code modularization and to semi-automatic user-construction of numerical PDE systems and initial mesh configurations for both Dirichlet or zero-Neumann boundary conditions. The most elementary types of node controlling penalty functions are obtained by inclusion in the minimization procedure of penalty terms $\sum(\varepsilon_i \dot{d}_i - S_i(d_i))^2$, where the d_i are triangle altitudes. The expressions for ε_i and S_i become extremely large (and thus restrict node motions) when the altitudes d_i approach a user-specified minimum value. (More general, gradient-dependent penalty functions will be considered in later 2-D work.) The evaluation of the inner products which appear in Equations (4) can be performed at each time step analytically for such simple terms as (α, α) and by numerical quadrature integrations otherwise. The primary information which is required for these inner product evaluations comes from the mesh geometry. The basic geometrical data structure, and hence the analytic structure, of the MFE equations are established once, and for all times, at the outset of code compilation--which thereby affords both conceptual simplicity and practical economies in MFE computations in 2-D. These features also make MFE codes well-suited for vector or parallel processing computers.

ODE Integration

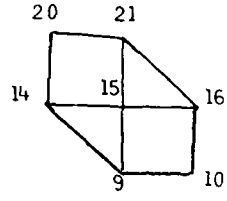
The stiffly stable numerical integration methods which can be used to solve the system of ordinary differential equations (4) and (5) usually contain a series of backward Cauchy-Euler steps interspersed with interpolations and extrapolations, all amidst error-controlling tests and strategies. A backward Cauchy-Euler (BCE) step is obtained by replacing $\dot{y}$ in Equation (5) by the backward difference $(y - \tilde{y})/\Delta t$, where $\tilde{y}$ is the known parameter vector at the previous time, and y is the unknown parameter vector at the current time. Upon linearization, the BCE equation then reads as

$$R(y, \tilde{y}) \approx R(\tilde{y}) + R'(\tilde{y}) \cdot \delta y = 0 \quad . \qquad (6)$$

The structure of the mass matrix C(y) determines in large measure the degree of difficulty and computational cost of obtaining numerical solutions for δy in Equation (6) and thus for the y array in Equations (4) and (5). Both an implicit Runge-Kutta method (DIRK2) and the Gear method are used interchangeably in the present test MFE code in 2-D; both of these stiff ODE solvers incorporate the backward Cauchy-Euler steps described above.

Matrix Solutions

The structure of the mass matrix C(y) can be seen by considering the 15th node in a 6 x 6 grid mesh of the type shown in Figures 1-4. The coupling among nodes in Equation (4) is represented schematically by



and the block structure of the mass matrix is shown in Figure 5. For a PDE system with 2 "problem variables," the segment of the dependent variable array y associated with the 15th node is $(y)_{15} = (a1_{15}, a2_{15}, x_{15}, y_{15})^*$, and the 4 x 4 block $C_{15,16}$ is

$$C_{15,16} = \begin{bmatrix} (\alpha_{16}, \alpha_{15}) & 0 & (\beta 1_{16}, \alpha_{15}) & (\gamma 1_{16}, \alpha_{15}) \\ 0 & (\alpha_{16}, \alpha_{15}) & (\beta 2_{16}, \alpha_{15}) & (\gamma 2_{16}, \alpha_{15}) \\ (\alpha_{16}, \beta 1_{15}) & (\alpha_{16}, \beta 2_{15}) & \sum_{k=1}^{2} (\beta k_{16}, \beta k_{15}) & \sum_{k=1}^{2} (\gamma k_{16}, \beta k_{15}) \\ (\alpha_{16}, \gamma 1_{15}) & (\alpha_{16}, \gamma 2_{15}) & \sum_{k=1}^{2} (\beta k_{16}, \gamma k_{15}) & \sum_{k=1}^{2} (\gamma k_{16}, \gamma k_{15}) \end{bmatrix} . \quad (7)$$

*The co-ordinate variable y_{15} of the 15th MFE node is not to be confused with the dependent variable of the MFE method which is also denoted by y.

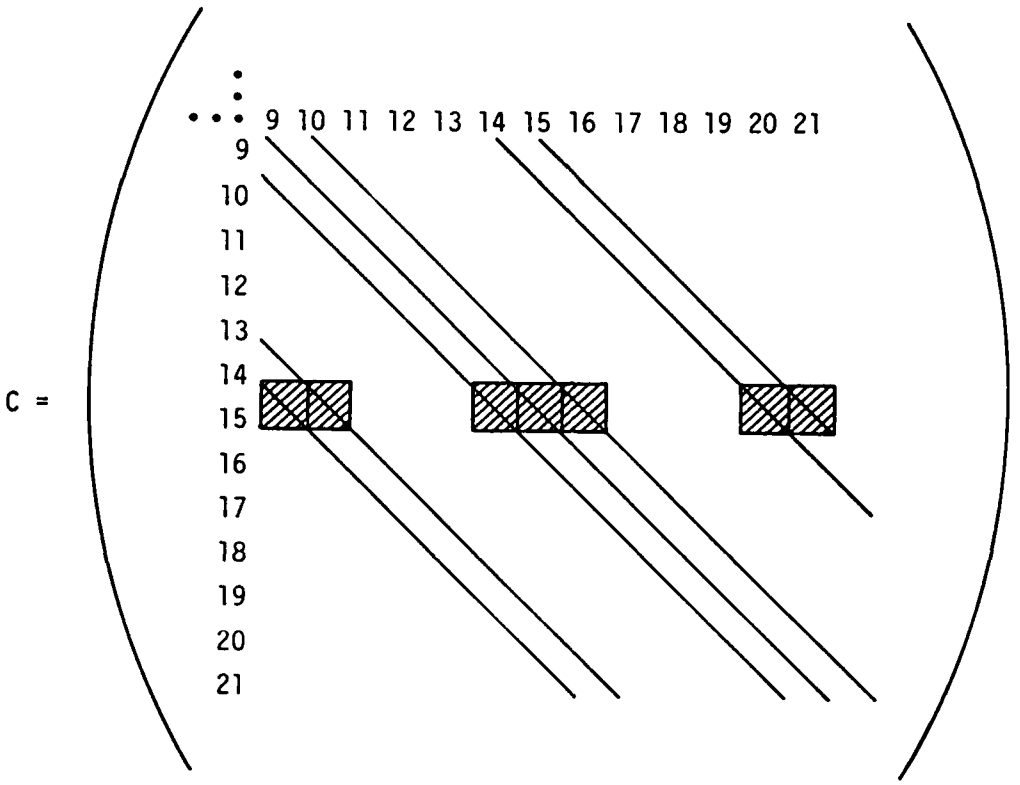


Figure 5. Block Structure of the Mass Matrix C(y).

Although direct L-U decomposition and solution of Equation (6) is relatively slow, it is reliable and is thus used in the initial code versions in the present research effort. More efficient matrix solution methods are in early stages of investigation and are showing some signs of promise for effective implementation in later stages of this study.

II. RESEARCH PROGRESS

Objectives

The overall objective of this research is to investigate the numerical properties and structure of the MFE method in 2-D in order to reduce it to practice for important PDE systems. In order to conduct the essential research of the MFE method in 2-D, a number of related objectives/technical milestones have been addressed and are summarized here with a brief comment on progress made during the first year's effort:

1. Experimental computer program. In order to conduct research efficiently on the MFE method in 2-D, an experimental computer program, which is entirely research oriented, is under development. Piecewise linear solution approximants defined on a hexagonally-connected triangular mesh were chosen for the initial 2-D code implementation. A highly modular and flexible code structure is being used in this work in order to facilitate investigations of numerous classes of PDE's which are of interest. An initial version of this experimental program is now executing MFE solutions successfully for heat equations, simple advection and Burger's equations, amply fulfilling this objective for the first year.

2. Node control methods. This milestone encompasses analysis of numerical and nodal behavior for conservation equations, with consideration of both advection-dominated and diffusion-dominated equations. Movement of the MFE nodes is controllable to a large extent by regularization terms in the MFE minimization procedures. Suitable regularization procedures (or "penalty functions") can ensure that the mesh cannot get tangled during the evolution of the problem. In 1-D research, it was seen that a small number of control parameters would allow a great deal of flexibility in the type of node mobility available in specific problems while satis-

fying a variety of logical and analytical criteria. For early developmental purposes, a simple regularization scheme which is similar in concept to the scheme used in 1-D has been incorporated in the 2-D code. Initial results in 2-D indicate that the additional degrees of freedom which are present in the 2-D node-moving equations have a beneficial effect on the MFE integration properties, vis a vis the more highly constrained nodal equations in only one spatial dimension.

3. Matrix solutions. The MFE method reduces coupled PDE's on the problem domain into a set of coupled ODE's on the nodes of a discrete, movable mesh. Numerical solution of these coupled ODE's then reduces to solutions of a matrix equation of rather high order for any reasonable number of nodes and of physical variables. For the triangulation which is presently under study in this 2-D MFE research, the Jacobian in the ODE problem is a banded matrix, with a relatively narrow band, permitting solution by L-U decomposition. Such a solution package has been coded and tested for the 2-D code. These solutions have also been verified against identical results from an IMSL package. We have also tested point and line relaxation methods and found them to converge too slowly for the production purposes which are ultimately envisaged. Testing of dynamic ADI methods has also been undertaken. Two-directional ADI appears to work well (if quadrilateral grid meshes are to be used), and three-directional ADI for a triangular grid mesh requires more research before conclusions can be made.

The first year's objectives have thus been satisfactorily met, with implementation of a first-generation 2-D code and related analyses and experiments which have demonstrated the feasibility of the MFE method in the 2-D problems tested to date. A substantive technical discussion of this progress appears in the following sections of this report.

STATUS OF THE RESEARCH

A. Experimental 2-D Computer Program

The fundamental architecture of an MFE code is determined largely by the type of solution approximants and the associated basis functions which are used. For reasons of simplicity and in view of the experience gained in

earlier 1-D research, piecewise linear solution approximants are used in this 2-D research. This selection of piecewise linear functions is further supported by numerous results in 1-D which confirm that PDE solution approximants of higher degree are not nearly so critical for attaining high levels of stability and accuracy in a moving node solution method as they are in fixed node solution methods. Given this selection of piecewise linear functions, alternative initial test code architectures in 2-D have been designed to operate either on a triangle-by-triangle basis or on a node-by-node basis. The properties of these respective code architectures are under intensive study because later applications to real-world problems in fluid dynamics or in continuum mechanics, for example, will require both node-by-node and triangle-by-triangle code evaluations.

A large amount of the computational effort in the MFE method is devoted to repetitive evaluations of the inner product terms in Equations (4), which are then used iteratively in the numerical ODE integration procedures. Because invariant geometrical relationships can be exploited frequently in the evaluation of inner products, these geometrical relationships are generated and encoded once, and for all time, from the initialization data at the outset of code calculations. Figures 6-8 illustrate the grid mesh conventions which would be established and encoded by the present 2-D code for 42 nodes on a 7 x 6 grid. The manner in which these relationships are used can be seen readily by considering a single conservation equation

$$u_t = -f_x - g_y + u_{xx} + u_{yy} \tag{8}$$

and the piecewise linear basis functions about the i^{th} node of the form

$$\alpha_i = a_i x + b_i y + c_i \tag{9}$$

The inner products (α_i, α_j) which appear in the mass matrix of Eqns. (4) are evaluated as

$$(\alpha_i, \alpha_j) = \int_{6T} dxdy\, (a_i x + b_i y + c_i)^2 , \tag{10}$$

where the i^{th} node is surrounded by six adjoining triangles T. The two-dimensional integration over a triangle T in Equation (10) can be performed either analytically or by the midpoint rule according to the formula

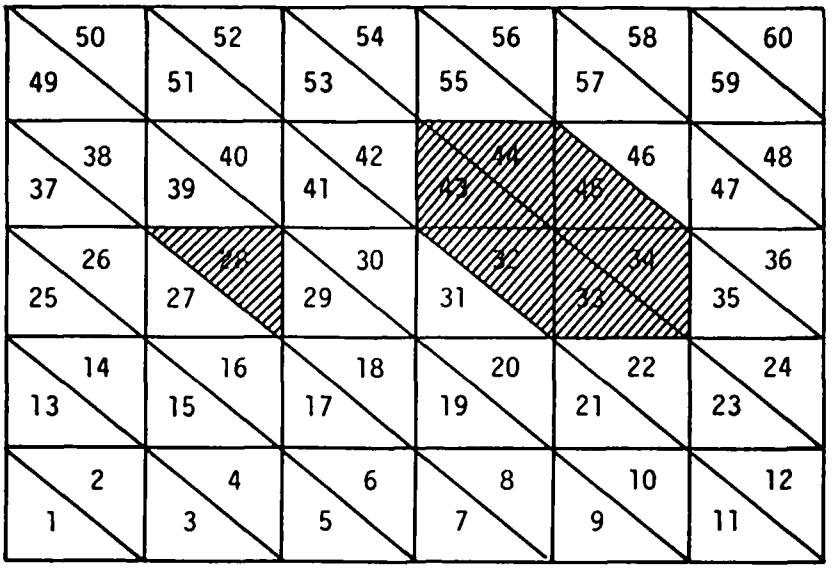


Figure 6. Triangles.

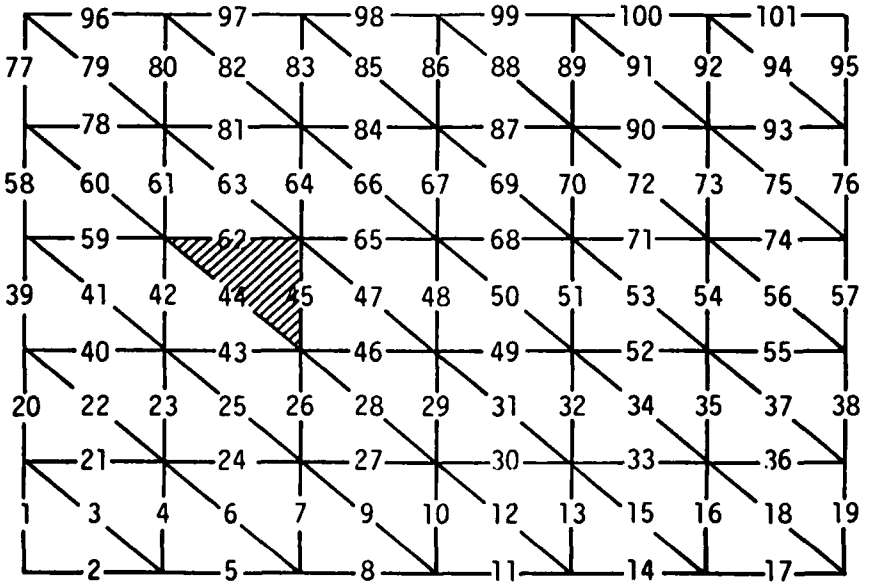


Figure 7. Edges.

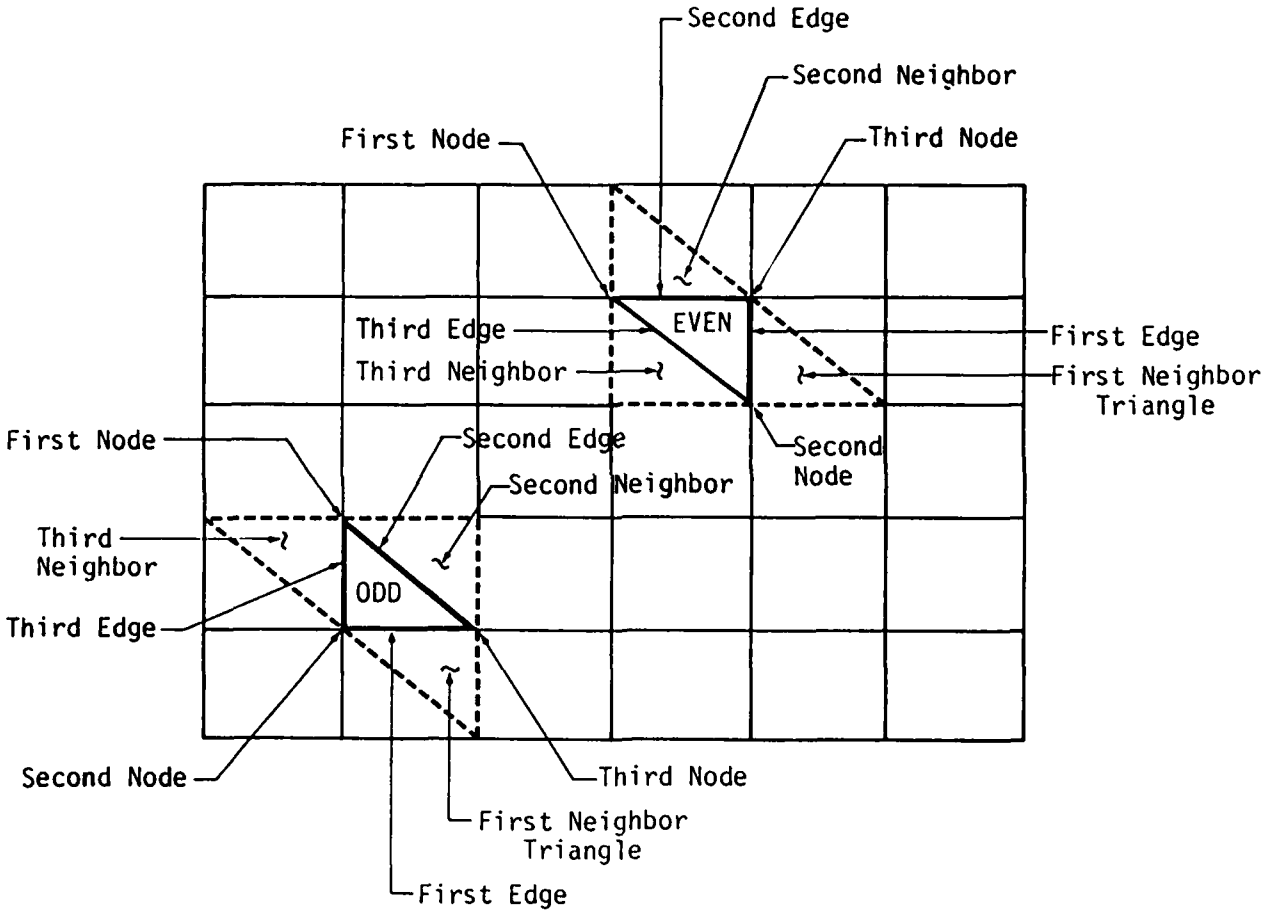


Figure 8. Triangle Conventions.

$$\int w(x, y)\, dxdy \simeq (A/3)\,[w(P_1) + w(P_2) + w(P_3)] \quad , \qquad (11)$$

where A is the area of triangle T, and P_1, P_2, and P_3 are the midpoints of the sides. This simple midpoint formula of integration is exact whenever the integrand, $w(x, y)$ is a quadratic function in x and y. The present research code uses the above midpoint rule and contains options for the use of a composite midpoint and some analytic integration methods, as well. Using the relationships between basis functions $\beta_i = u_x\alpha_i$ and $\gamma_i = u_y\alpha_i$, and noting that u_x and u_y are constants, the inner products (α_i, β_i), (β_i, β_i), (β_i, γ_i) and (γ_i, γ_i) can also be evaluated readily. For a general operator $L(u) = f$, it can be shown that

$$(-f_x, \alpha_i) = \sum(\alpha_x)_i \int dxdy\, f \quad , \qquad \text{and}$$

$$(-f_x, \beta_i) = \sum u_x \left\{-(\alpha_x)_i \int dxdyf + \sum_{\substack{2 \text{ inner} \\ \text{edges of T}}} n_1 \int f\, \tau(s)ds \right\} .$$

The integration on ds is performed with respect to arc length s along a triangle edge and $\alpha_i = \tau(s)$ is the linear function of s which rises from a value 0. at an outer node to a value 1.0 at the ith central node. Edge integrals can be evaluated analytically or by numerical quadrature. Simpson's rule and a composite Simpson's rule are available for use in the present research code. The derivatives α_x, u_x, and the x component of the unit outward normal n_1 are readily determined at all times by invariant formulae from the known amplitudes and nodal co-ordinates at triangle vertices.

Inner products of Laplacian operators are more complicated to evaluate, requiring, in addition to the quantities above, the unit tangent vectors and their x and y components.(7) The formulae for these quantities are also encoded at the outset of computations for repeated use in later code calculations.

The overall 2-D code structure is conceptually simple, as indicated in Figure 9. The detailed code structure contains an assembly of approximately forty subroutines which perform modularly the numerous subtasks which are

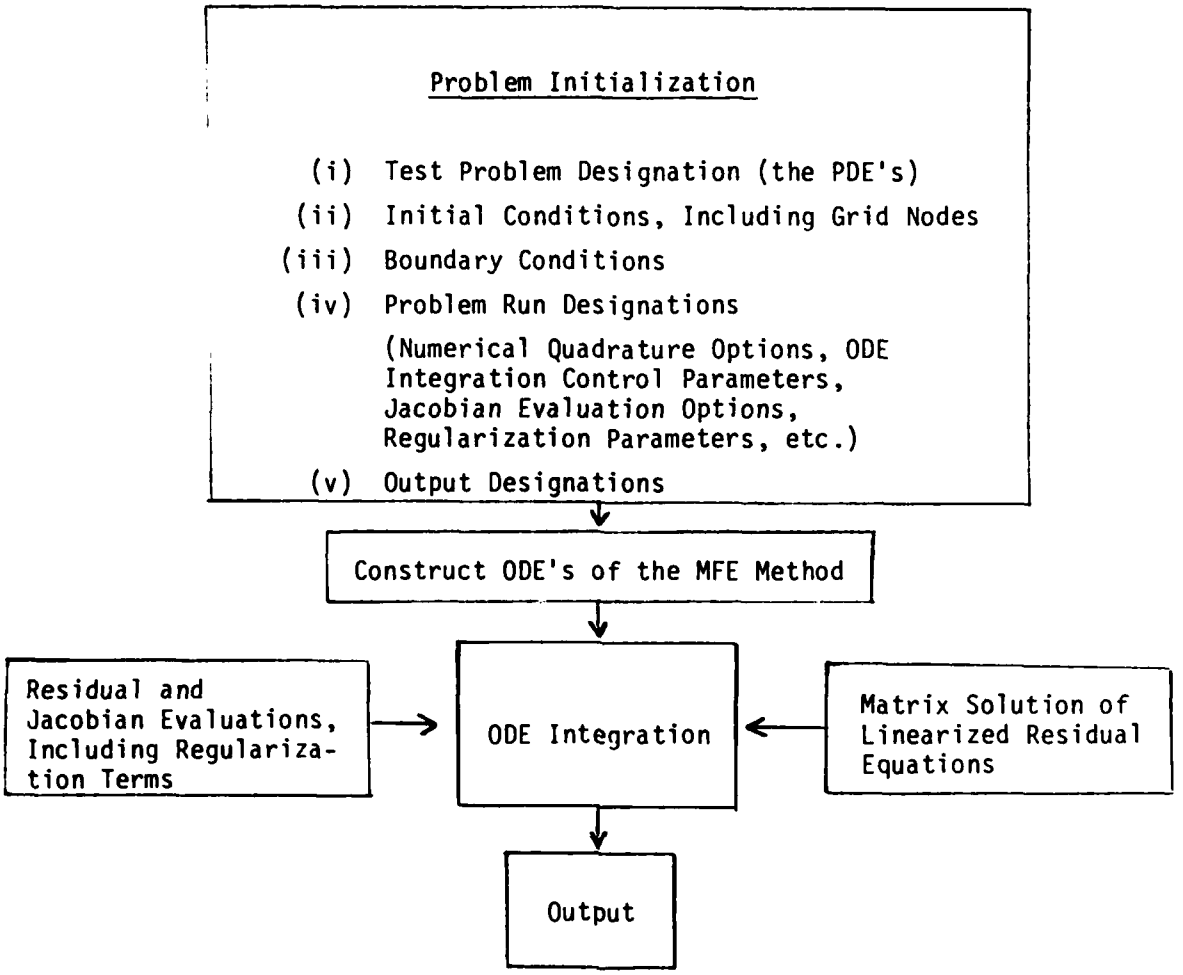


Figure 9. Schematic Representation of Major Functions Performed in the Present MFE Codes in 1-D and 2-D.

required in order to execute the major code functions indicated in Figure 9. This structure facilitates mathematical research on alternative (actually, interchangeable) ODE solution methods, matrix solution methods, node control strategies, boundary conditions, and such other tasks as grid mesh generation and issues of numerical analysis which must be investigated in new PDE solution methods. This code structure of the MFE method also appears to be highly compatible with vector and parallel processing computers.

B. Node Control Methods

Because the mass matrix $C(y)$ in Equation (5) can become singular (whenever all components u have a flat portion in their graph at some node (x_i, y_i)), regularization terms are included in the minimization problem for the x and y equations. The minimization problem for $\dot{x}$ and $\dot{y}$ reads $\frac{\partial \psi}{\partial \dot{x}_k} = 0$ and $\frac{\partial \psi}{\partial \dot{y}_k} = 0$, where

$$\psi \equiv \sum_i w_i^2 \, || \, \dot{u}_i - L_i(U) \, ||^2 + \sum_{\substack{\text{triangle} \\ \text{altitudes}}} (\varepsilon_j(d_i) \, \dot{d}_j - S_j(d_j))^2 \quad . \tag{12}$$

This minimization then yields regularization terms of the form:

in the $\dot{x}$ equation

$$\varepsilon_j \{\varepsilon_j \dot{d}_j - S_j(d_j)\} \cdot \frac{\partial \dot{d}_j}{\partial \dot{x}_k} \tag{13a}$$

in the $\dot{y}$ equation

$$\varepsilon \{\varepsilon_j \dot{d}_j - S_j(d_j)\} \cdot \frac{\partial \dot{d}_j}{\partial \dot{y}_k}$$

It follows that the terms $\dot{d}_j = \frac{\partial d}{\partial x_k} \dot{x}_k + \frac{\partial d}{\partial y_k} \cdot \dot{y}_k$ and the terms $\frac{\partial \dot{d}_j}{\partial \dot{x}_k}$ and $\frac{\partial \dot{d}_j}{\partial \dot{y}_k}$ in Equations (13) are dependent primarily upon grid geometry; and their invariant formulae are thus generated at the outset and encoded for repeated calculations later in the code in a manner similar to many portions of the residual evaluations which were discussed above.

The functions ε^2 and εS which appear in Equations (13) are treated in exactly the same manner as an advanced penalty function formulation which has been tested previously in 1-D investigations(7) of effective regularization methods; that is,

$$\varepsilon^2 = \frac{C_1}{(d - SEPMIN)} \tag{14}$$

$$\varepsilon S = \frac{C_2}{(d - SEPMIN)^2} \quad , \tag{15}$$

where SEPMIN is a minimum triangle altitude. A detailed node controlling policy has been developed for the selection of values for the constants C_1 and C_2 in terms of local truncation errors and of the inner products (β, β) and (γ, γ) which appear in the $\dot{x}$ and $\dot{y}$ equations. A discussion of this policy in specific 1-D problem applications appears in Reference 6. A detailed discussion of this policy for 2-D applications is deferred until several additional generalizations which are presently under development in 2-D are completed. These generalizations will also have the effect of preventing automatically the inversions (tangling) of triangles. Such inversions are presently detected and prevented by testing the aspect ratios and the signs of triangle determinant quantities. Integration time steps are reduced when aspect ratios exceed a designated value or when there is a change of sign in the determinant quantities.

C. Matrix Solution

The BCE equations (6) are solved by Newton's iteration method. A single iterate of the linearized equations is expressed as

$$R(\bar{y} + \delta y) \simeq R(\bar{y}) + R'(\bar{\bar{y}}) \cdot \delta y = 0 \quad , \tag{16}$$

where $\bar{y}$ is the latest Newton iterate; $\bar{y} + \delta y$ is the next iterate, and $\bar{\bar{y}}$ is the latest value of y which has been used to evaluate the Jacobian R'. (Due to the large expense of calculating Jacobians in 2-D, a modified Newton's method is also used whenever possible.) Upon introducing the variables $z \equiv (y-\bar{y})/\Delta t$ and

$$R(y) = F(y, z) \equiv C(y) \cdot z - g(y) \quad , \tag{17}$$

it follows that

$$R'(y) = F_y + F_z \cdot z_y = \frac{\partial F(y,z)}{\partial y} + \frac{C(y)}{\Delta t} \quad . \tag{18}$$

The resulting matrix equation,

$$\frac{C(\bar{\bar{y}})}{\Delta t} + \frac{\partial F(\bar{\bar{y}},\bar{\bar{z}})}{\partial y}\, \partial y = - R(\bar{y}) \quad , \tag{19}$$

can then be solved for δy by numerous matrix solution methods. The Jacobian $\partial F/\partial y$, can be evaluated either analytically or by numerical incrementation. Two options of numerical incrementation are available in the present 2-D code; the first option employs a central difference method, and the second option employs a forward difference method. This is apparently not a critical choice because small errors in Jacobians usually affect only the rate of convergence--and not the stability and eventual accuracy of the MFE solutions.

Several matrix solution methods have been tested in this work for use in the 2-D MFE method, including both point and line relaxation methods, alternating direction implicit (ADI) methods, and (direct) L-U decomposition methods. It was found that the point and line relaxation methods converge too slowly to be effective in MFE calculations in 2-D. Two-directional dynamic ADI was found to be quite effective in its own right for a quadrilateral grid, and it may also be useful as a preconditioning method for such other matrix solution methods as conjugate gradient methods. The triangular mesh which is used in the present MFE work in 2-D would require, however, a three-directional ADI method for solution of the matrix equations (19). Although promising, the three-directional ADI method still requires continuing research on issues of definiteness prior to implementation as a matrix solution method in the MFE method. Matrix reduction methods which are based upon multi-grid techniques provide a promising alternative to ADI for the triangular mesh used in this work. These reduction methods are in only the early stages of research for possible use in the MFE method, and they will be investigated more extensively in the coming year.

In view of its reliability, and notwithstanding its relatively large computational overhead, an L-U decomposition method is presently used to solve the matrix equations (19). With Equations (19) written in the form Ax

= B, where $x \equiv \delta y$, etc., the matrix A can be decomposed into a lower triangular matrix L and an upper triangular matrix U in which the values 1.0 appear (arbitrarily) on the diagonal of U. The system LUx = B is then solved successively, using Gauss elimination, in two stages: first, the system Lm = B is solved for m, and second, the system Ux = m is solved for $x = \delta y$. Testing of this linear solver on independent test problems and comparisons to results from a similar banded matrix solver package in the IMSL library on the Lawrence Berkeley Laboratory computer system comfirm the reliability of the direct L-U decomposition method for yielding accurate solutions of the BCE equations (19).

D. Test Problems

Numerous simple test problems have now been used for initial testing of the MFE method in 2-D. These problems have been designed to test such numerical aspects of the MFE method as:

(1) Inner product evaluations (analytic and numerical quadrature)
(2) ODE integration for the MFE equations
(3) Matrix solution methods
(4) Regularization schemes for control of node motions
(5) Jacobian evaluations
(6) Boundary conditions (zero-Neumann and Dirichlet).

The PDE's in these test problems are in the form of general conservation equations,

$$u_t = -f_x - g_y + \nu(u_{xx} + u_{yy}) \ , \qquad (20)$$

where the flux functions f and g can have nearly arbitrary functional forms.

In addition to the simple examples of the heat equation and of square wave propagation which were presented in Reference 7 , two somewhat more complex test examples have also been studied.

(i) Oblique Propagation of a Scalar Wave

This example considers the propagation by pure advection of a scalar wave diagonally across the initial grid mesh, according to the equation $u_t + u_x + u_y = 0$.

The initial conditions for this example are shown schematically in Figure 10 and are expressed as:

$u(x,y,0) = 1.0$ $\quad 0.1 \leq x \leq 0.2; \quad 0.1 \leq y \leq 0.2$

$u(x,y,0) = 0.$ $\quad x \leq 0.05; \quad x \geq 0.25$ and all y

$u(x,y,0) =$ linear $\quad$ otherwise .

Dirichlet conditions, $u(x,y,t) = 0.$, are applied at all boundaries. The co-ordinates x and y obey the following Dirichlet conditions: $x = 0.$ along the y axis; $x = 1.0$ along the boundary $x = 1.0$, all y; $y = 0.$ along the x axis; and $y = 1.0$ along the boundary $y = 1.0$, all x. The co-ordinate variables obey zero-Neumann conditions on y along the y axis and along the boundary at $x = 1.0$ for all y and on x along the x axis and along the boundary at $y = 1.0$ for all x. (That is, the y co-ordinate is free to slide along the vertical boundaries, and the x co-ordinate is free to slide along the horizontal boundaries.)

Using a 6x6 grid of moving nodes, this problem is run from $t = 0.$ to $t = 0.8$, which spans the interval of free propagation. The accuracy of the wave profile is maintained to within 1 part in 10^3, consistent with the local truncation error constraint in the ODE solver. The velocity of propagation is accurate to 4 significant figures. The mesh is observed to move flexibly in order to maintain these accuracies throughout the problem evolution. As this wave approaches the upper right-hand corner of the domain, aspect ratios of some of the grid mesh triangles approach values on the order of approximately 10^2, with no adverse effects. The triangle aspect ratios can be made to assume values which are an order magnitude larger by imposing Dirichlet conditions on the x and y coordinates at the boundaries. It was also found that the MFE method can solve this problem with equal facility and efficiency for much larger, finite gradients of the scalar wave front.

This problem can readily be modified so that the scalar wave trajectory follows a circular path according to the pure advection equation,

$$u_t = \cos(t) \cdot u_x + \sin(t) \cdot u_y \quad , \qquad -a < x < a \quad -a < y < a \quad , \qquad (21)$$

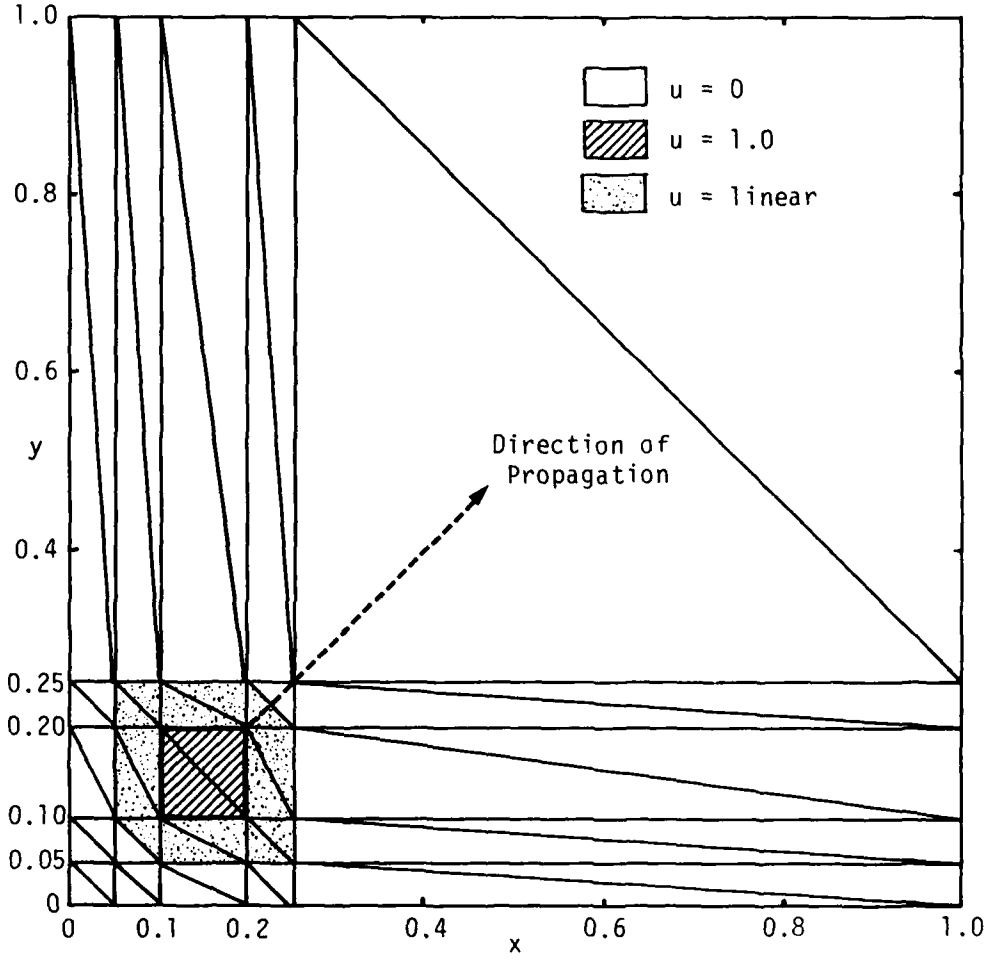


Figure 10.
Initial Conditions for Oblique Propagation of Scalar Wave.

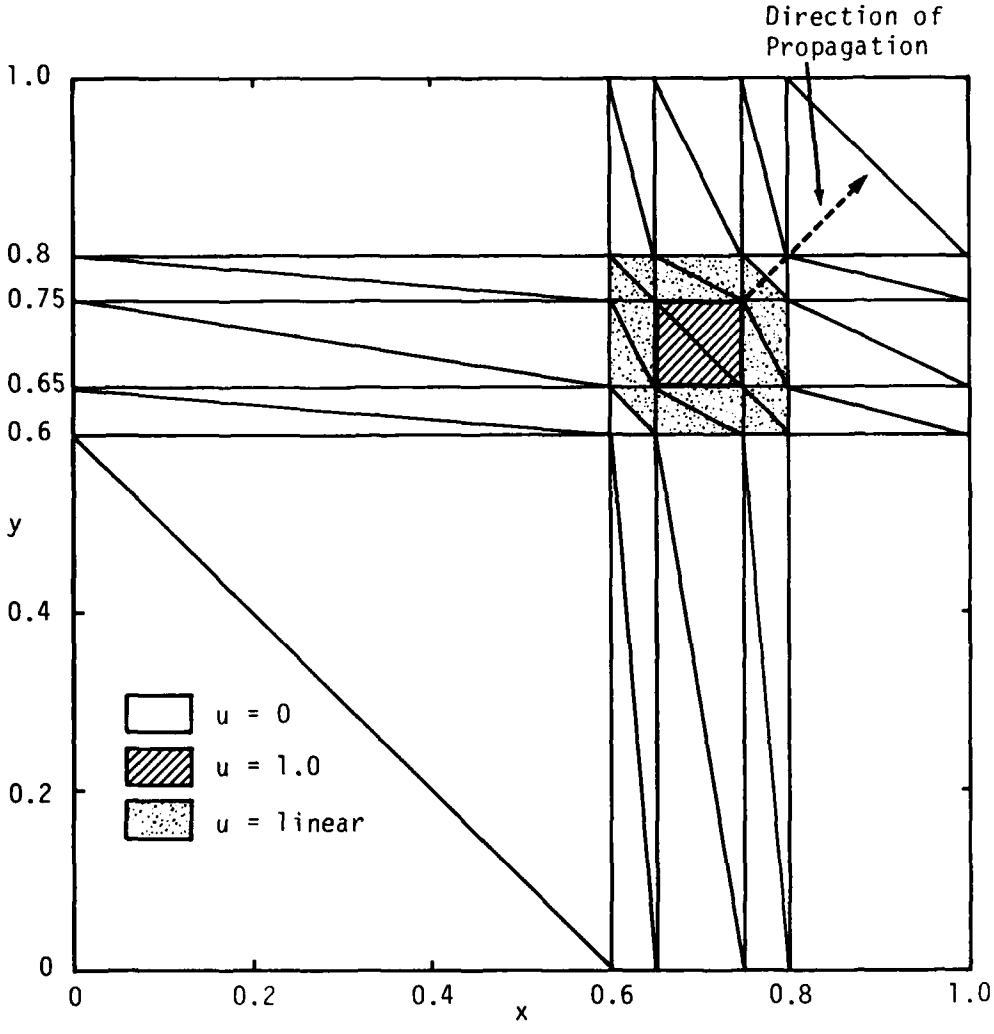


Figure 11.
Transient Propagation of a Scalar Wave.

where the dimensions a are sufficiently large to contain the circular trajectory. In this MFE solution, the scalar wave follows its circular trajectory accurately and returns precisely to its initial position, consistent with the local truncation error constraint in the ODE solver, after a complete revolution ($t = 2\pi$), using 20 time step cycles. The grid mesh again expands and contracts smoothly and flexibly in maintaining the desired solution accuracy at all times.

(ii) Burger-Like Equations.

A 2-D analog of Burger's 1-D model equation is given by the equations

$$\frac{\partial u}{\partial t} = -\frac{\partial (u^2)}{\partial x} - \frac{\partial}{\partial y}(uv) + \nu \nabla^2 u \qquad (22)$$

$$\frac{\partial v}{\partial t} = -\frac{\partial}{\partial x}(uv) - \frac{\partial}{\partial y}(v^2) + \nu \nabla^2 v \quad ,$$

where u and v can be viewed as x and y components of a fluid velocity, respectively. In order to maintain a close analogy to the 1-D Burger's model results which were discussed in Reference 2, initial and boundary conditions for this system of PDE's are first selected so as to lead to the propagation of a uniform, step-like wave in a direction parallel to the y-axis; that is,

$u(x,y,0) = 0.$ $\quad 0. \leq x \leq 1.0; \quad 0. \leq y \leq 1.0$

$v(x,y,0) = 1.0$ $\quad 0 \leq y \leq 0.100; \quad 0 \leq x \leq 1.0$

$v(x,y,0) = 0.$ $\quad 0.101 \leq y \leq 1.0; \; 0 \leq x \leq 1.0$

$v(x,y,0) = \text{linear}$ $\quad$ otherwise

This problem is solved from t = 0 to t = 0.5 on a 4 x 19 grid with $\nu = 10^{-2}$.* The dependent variable v obeys zero-Neumann conditions along the y axis and along the (vertical) boundary at x = 1.0; and v obeys the Dirichlet conditions, v = 1.0 along the x axis and v = 0. along the (horizontal) boundary at y = 1.0. All interior nodal co-ordinates obey the same type of sliding boundary conditions as were used in the scalar wave example discussed above. Figures 12-15 present the MFE solutions of this evolving wave front. The extensive migration of the MFE nodes from their initial positions to those positions which resolve the waveform at t = 0.2 is clearly evident in Figures 12 and 13. The speed of propagation and the shock-like wavefront solutions are resolved to accuracies of three significant figures, or better, consistent with the local truncation error constraint of the ODE solver. The magnitudes of the wavefront gradients are approximately 100 in this example, and MFE solutions of this problem can be obtained with similar facility and efficiency for much larger gradients (corresponding to smaller values of ν in Eqns. (22) and (23)). Consistent with earlier results in Reference 7 for simpler square wave propagation by pure advection, it is found also in this Burger-like example that wide latitude can be exercised in the selection of node controlling parameters in the functions ε^2 and εS of Eqns. (14) and (15). When relatively weak internodal forces are used, no nodal deviation (or bias) in the x direction is observed. (There are no transverse forces in the PDE's, per se.) As the internodal forces are increased to sufficiently large values, the nodes can be forced by the regularization terms to migrate toward the x-axis for the nodal triangulation shown in Figure 12--meanwhile maintaining the accuracies mentioned above. The direction of such forced nodal migration can be reversed by using the opposite type of grid triangulation, and this node migration can also be altered by the use of different penalty functions. Node control is thus very flexible and desired accuracies are readily maintained. When PDE's contain non-trivial x-dependencies in

---

*This problem can be solved with equal effectiveness on an MFE grid of 4x10 nodes. The 4x19 grid simply represents the initial attempt on this problem. The figures 12-15 below have rotated the co-ordinate axes in the x-y plane by 90° from the conventional orientation ($\hat{x}$ horizontal and $\hat{y}$ vertical) in order to improve the viewing angle for the results plotted in 3-D. The terms "horizontal" and "vertical" refer strictly to the conventional orientation of the x-y plane throughout this discussion.

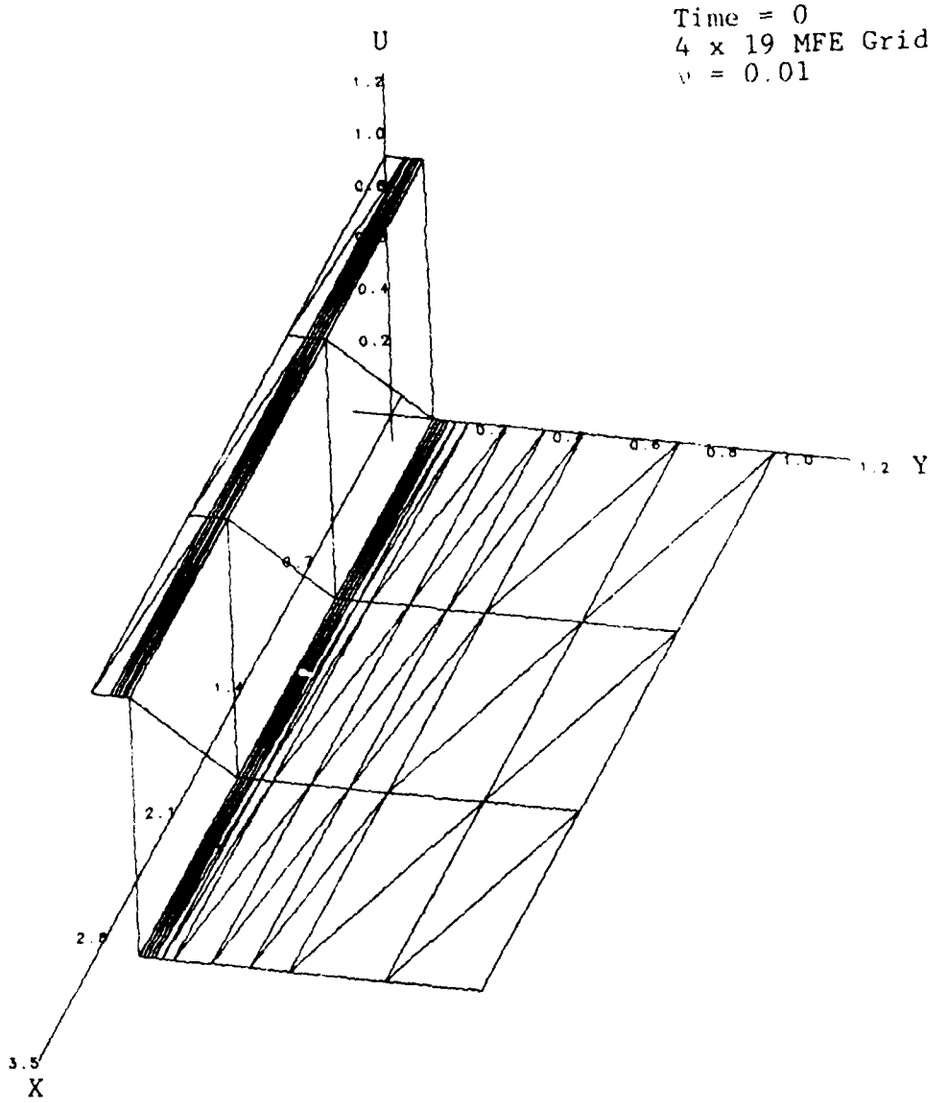


Figure 12.

MFE Solution of
Burger-like Equation in 2-D

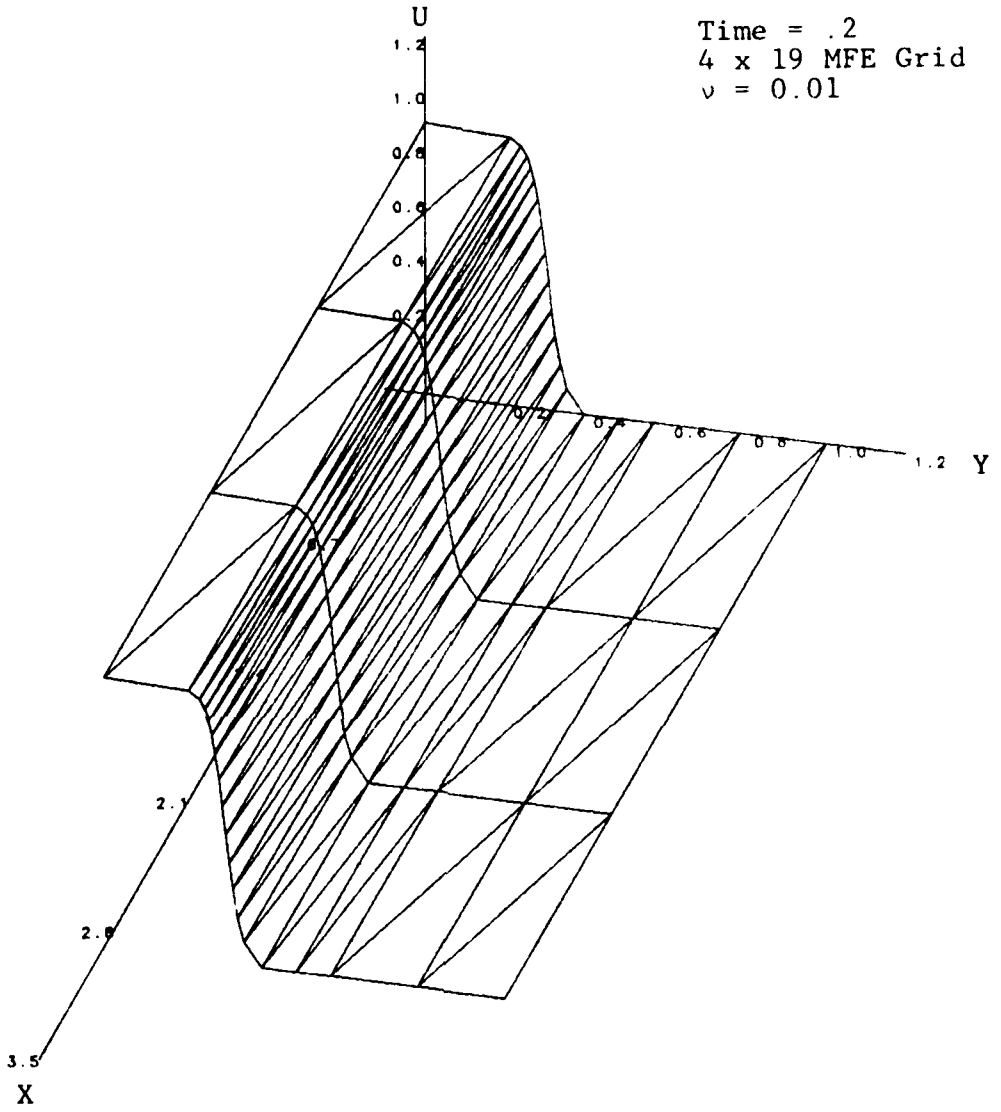


Figure 13.

MFE Solution of

Burger-like Equation in 2-D

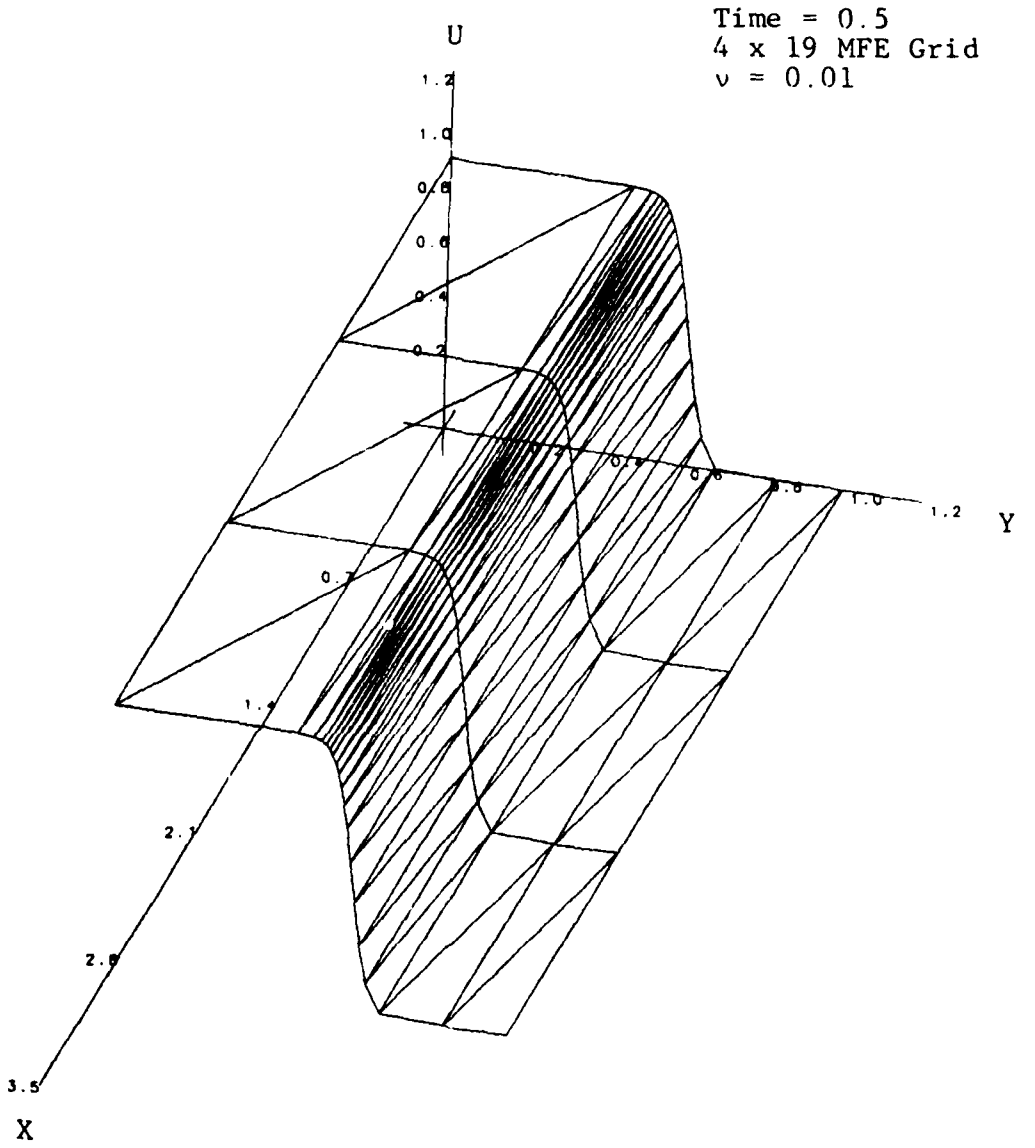


Figure 14.

MFE Solution of

Burger-like Equation in 2-D

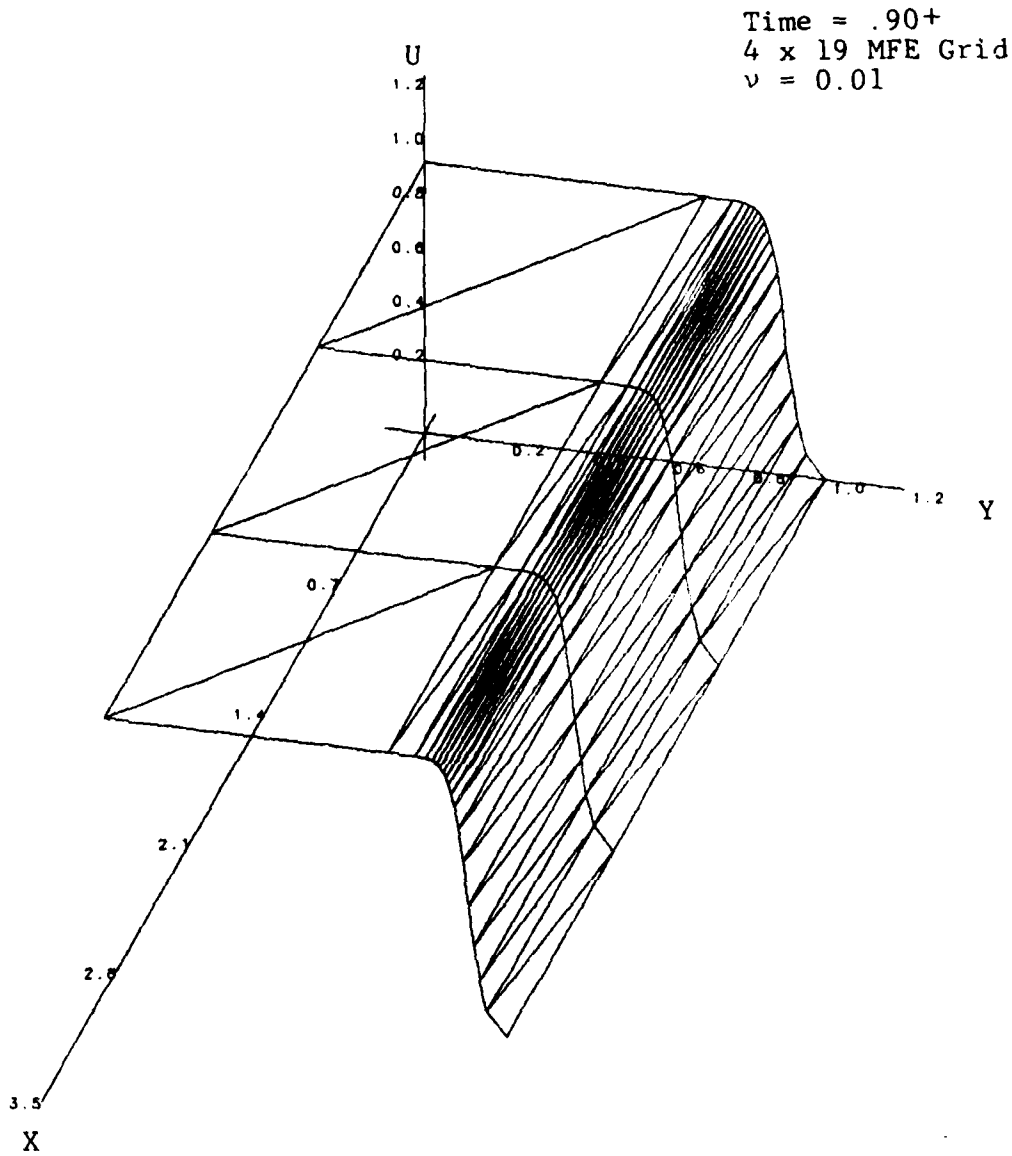


Figure 15.

MFE Solution of

Burger-like Equation in 2-D

their operators, the PDE's themselves resume their dominant role in governing the positioning of the nodes, as will be seen in a skewed waveform example below. Figure 15 shows the formation of the boundary layer at the right-hand boundary, as this MFE solution approaches the correct asymptotic solution.

Finally, an example of a skewed, propagating wavefront (shock) can be formulated in terms of the Burger-like equations,

$$u_t = 0 \tag{24}$$

$$v_t = -1/2\ (v^2)_y + \nu\nabla^2 v\ , \tag{25}$$

on the unit square. An 8x8 grid of MFE nodes is used, and the initial conditions for u and v on uniformly spaced grid nodes are:

$u(x,y,0) = 0$. all x,y

$v(x,y,0) = 1+7\tau,\ 1+6\tau,\ 1+5\tau,\ldots,1$ at nodes 1, 2, 3,...,8 along 1st row (x-axis)

$v(x,y,0) = -1,\ -(1+\tau),\ -(1+2\tau),\ldots,-(1+7\tau)$ at nodes 57, 58, 59,...,64 along top row.

The initial values of v(x,y,0) along a given vertical line are obtained by linear interpolation at interior nodes. The parameter τ is assigned a constant value of 0.01, and the value of ν in Eqns. (24)-(25) is assigned a value of 0.01 in the present run. As shown in Figure 16, these initial conditions on v simply map a plane in which v has the values +1.07 at the coordinates (0,0); +1.0 at (1,0); -1.0 at (0,1); and -1.07 at (1,1). Dirichlet boundary conditions are maintained on v(x,y,t) and on x and y along the horizontal boundaries; zero-Neumann boundary conditions are applied to v(x,y,t) and to y on the vertical boundaries; and Dirichlet boundary conditions are maintained on x along the vertical boundaries. This skewing of the initially counter-directed velocity components along the top and bottom boundaries leads to the evolution of non-uniform wavefront solutions which are seen in the results below. In the early stages of solution, prior to $t \tilde{=} 0.5$, a projection of the MFE solution on the x-y plane shows two counter-directed,

quasi-horizontal wave impulses which propagate from top to bottom and from bottom to top at speeds of approximately 0.5. At t = 0.5, a shock is formed when the propagating impulses encounter each other near the horizontal center-line of the x-y domain. Subsequently, a skewed, shock-like waveform is generated and propagates in the serpentine manner shown in Figures 16-19. The relatively large aspect ratios seen in Figure 19 for the MFE mesh at t = 20. were created deliberately by the use of Dirichlet boundary conditions on the x co-ordinates along the x axis and along the parallel boundary line at y = 1. The MFE node migration was fluid throughout and exhibited no grid-biasing effects. This problem was run from t = 0. to t = 20. in approximately 125 time-step cycles. The gradients of the fully developed shock are on the order of 100, consistent with the present value of $\nu = 0.01$. As above, MFE solutions of this problem on an 8x8 grid can be obtained for much larger gradients (smaller values of ν) with essentially the same levels of robustness and efficiency as are seen in the present example.

From the results obtained in 2-D to date, it is apparent that the hexagonally connected triangular mesh used here and perhaps several other possible triangulation schemes are quite compatible with the MFE method. The additional degrees of geometrical freedom which are available for error minimizing node motions in 2-D have been found, so far, to have a beneficial effect on the numerical integration efficacy of the MFE method in 2-D, vis a vis the more highly constrained nodal motions in 1-D MFE solutions.[2,3] The nodal movement properties observed in these 2-D examples thus suggest some likely implications on the eventual role of mesh generation needs and data structure issues in the MFE method. So long as the MFE method continues to exhibit this robustness in its continuous node movement properties, the use of grid mesh generation routines would be reduced primarily to problem initializations. This anticipated restriction to problem initializations thus minimizes the likely role of, and demands upon, conventional mesh generation methods which may be used in conjunction with the MFE method. It is, of course, possible that some infrequent remapping of MFE solutions may sometimes be required in order to reach final solutions; in such circumstances, one would undoubtedly choose to interrupt the MFE solution, regrid the numerical PDE solution data at that stage, and then proceed again with the MFE solution as a new initial-value problem. For this type of procedure, the data structures would remain invariant in each individual stage of numerical solution, and the MFE method,

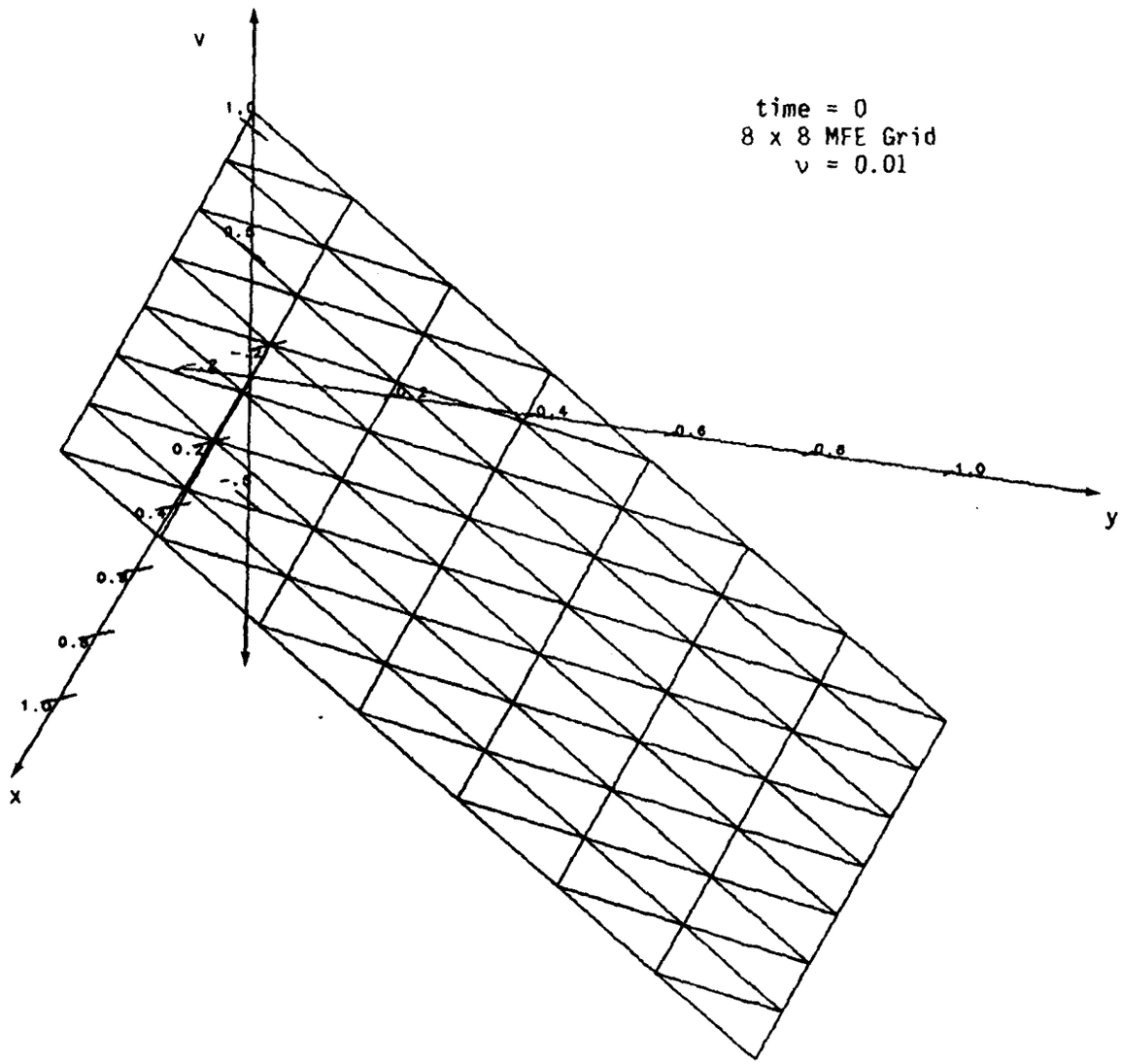


Figure 16. MFE Solution of Burger-Like Equations for a Skewed Waveform in 2-D.

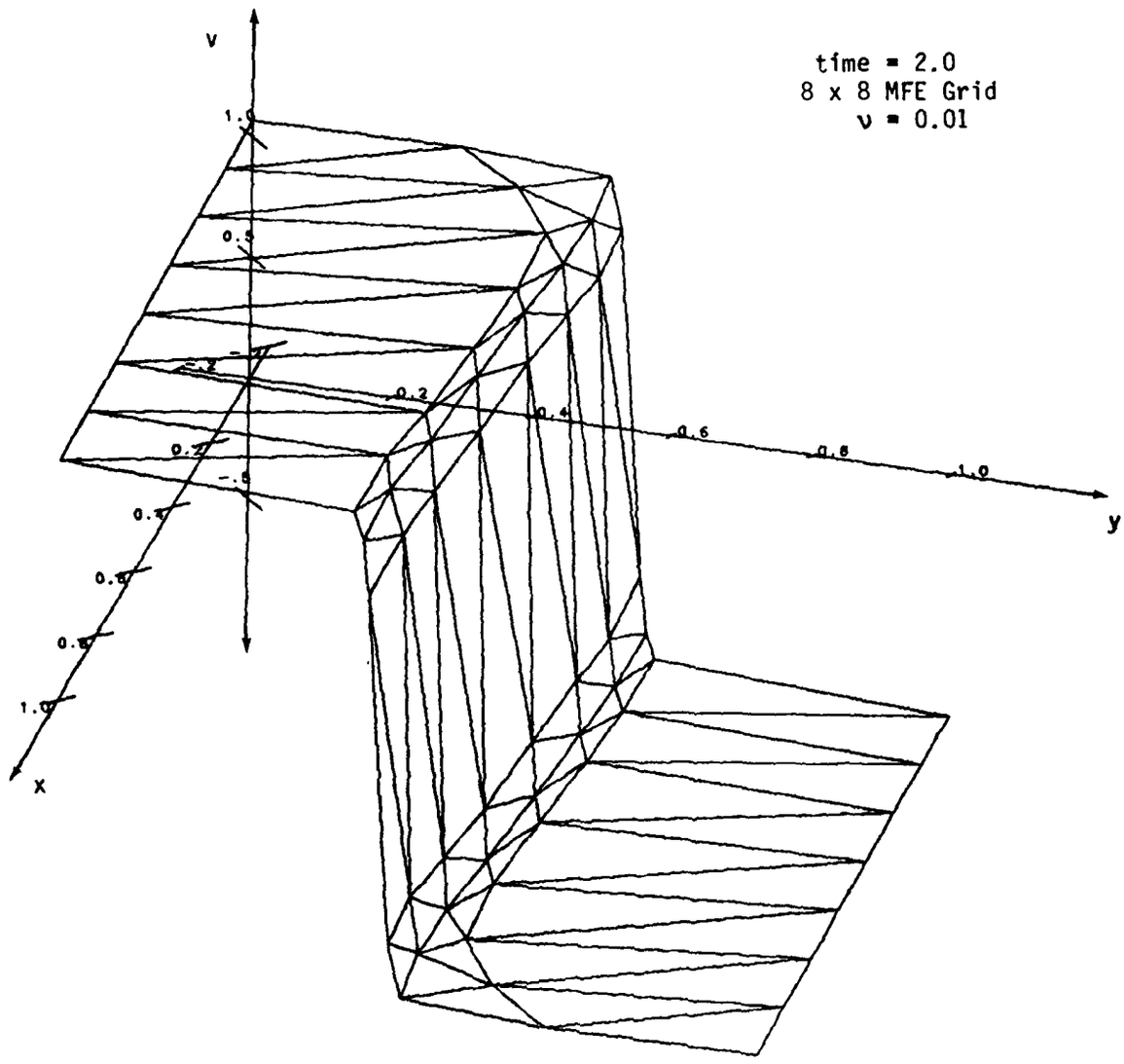


Figure 17. MFE Solution of Burger-Like Equations for a Skewed Waveform in 2-D.

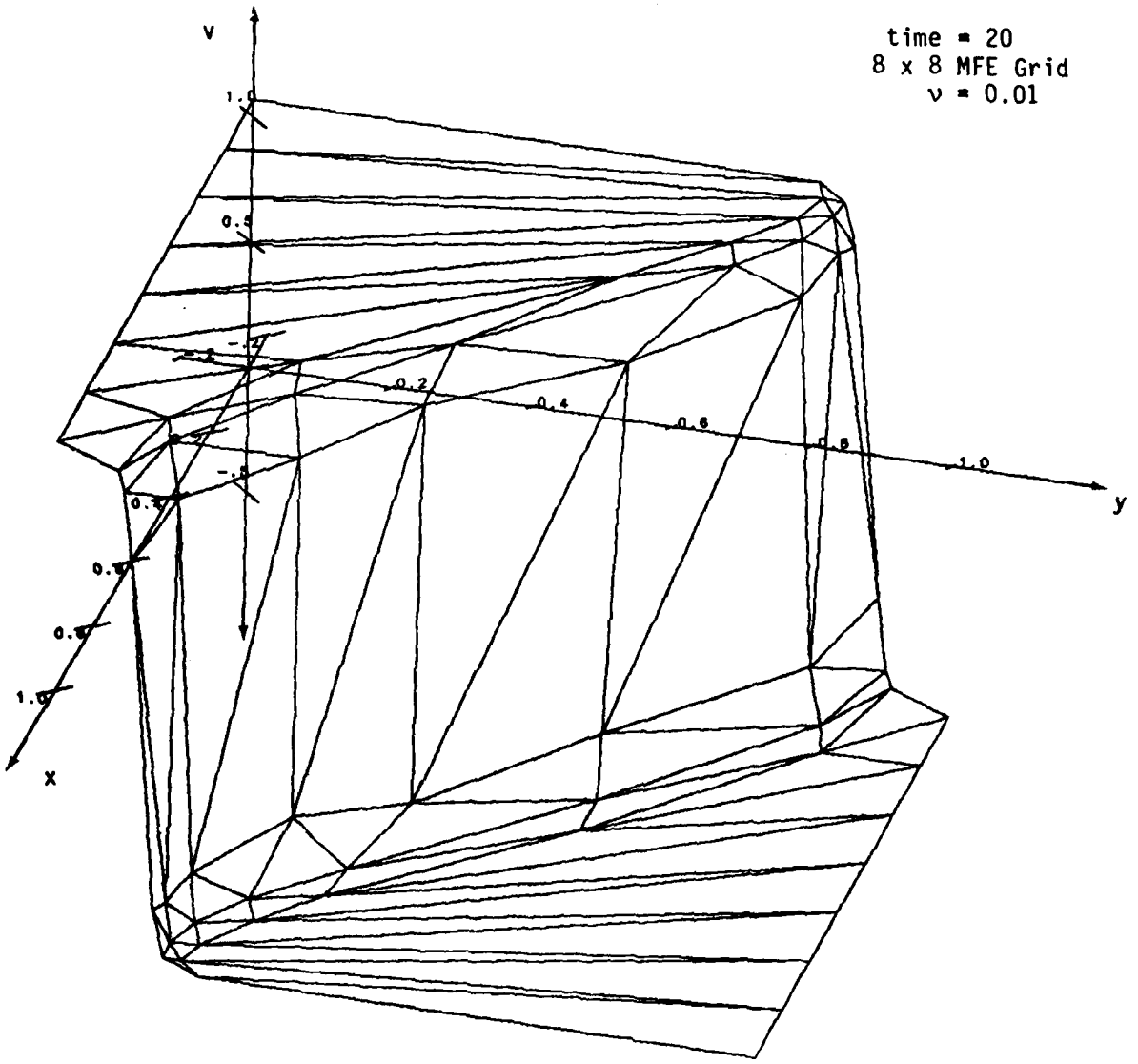


Figure 18. MFE Solution of Burger-Like Equations for a Skewed Waveform in 2-D.

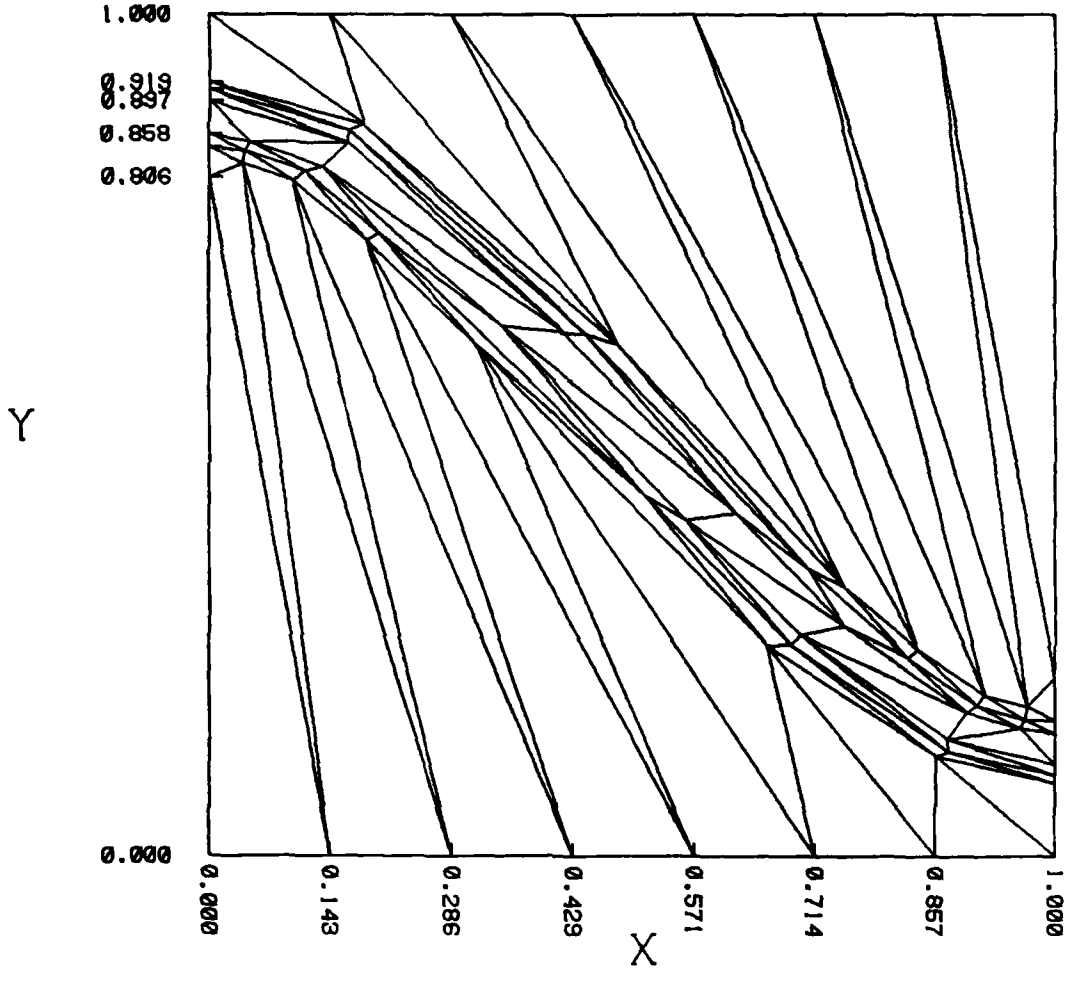


Figure 19. Projection on the x - y Plane of the MFE Solution of Burger-Like Equations for a Skewed Waveform in 2-D. time = 20.; 8 x 8 MFE Grid; ν = 0.01.

per se, would continue to serve as the dynamic mesh generator. It thus appears that, if any revisions at all are required in existing mesh generation methods for MFE initializations, only minor alterations of mesh indexing and of triangle orientations would be needed.

IV. ADDITIONAL INFORMATION

A. JOURNAL PUBLICATIONS

1. Gelinas, R.J., Doss, S.K., and Miller, K., "The Moving Finite Element Method: Applications to General Partial Differential Equations with Multiple Large Gradients," J. Comp. Phys., 40, No. 1, p. 202 (March 1981).

2. Prosnitz, D., Haas, R.A., Doss, S.K., and Gelinas, R.J., "Two-Dimensional Numerical Model Of a Free Electron Laser Amplifier," to appear in the J. of Quantum Electronics. (Summary presented at the Conference on Lasers and Electro-Optics (CLEO-81), Washington, D.C., June 10-12, 1981.)

B. MANUSCRIPT IN PREPARATION

1. Gelinas, R.J., Doss, S.K., Vajk, J.P., Djomehri, M.J., and Miller, K., "The Moving Finite Element Method: 2-D Applications," to be submitted to J. Comp. Phys., in late 1982.

C. PROFESSIONAL PERSONNEL ASSOCIATED WITH THE RESEARCH EFFORT

Dr. M. Jahed Djomehri (student of Prof. Keith Miller. Dr. Djomehri, who has a Ph.D. in Nuclear Engineering, will receive a Ph.D. in Mathematics at U.C. Berkeley during the summer of 1982.

Dr. Said K. Doss

Dr. Robert J. Gelinas

Dr. J. Peter Vajk

D. INTERACTIONS (COUPLING ACTIVITIES)

(i) Papers presented at meetings and conferences

1. Gelinas, R.J. and Doss, S.K., "The Moving Finite Element Method: A Semi-Automatic Solver for Diverse PDE Applications," Advances in Computer Methods for Partial Differential Equations - IV, pp. 230-239, Edited by R. Vichnevetsky and R.S. Stepleman, Proceedings of the Fourth IMACS International Symposium on Computer Methods for Partial Differential Equations held at Lehigh University - Bethlehem, PA, U.S.A., June 30 - July 2, 1981.

2. Vajk, J.P., Doss, S.K., Gelinas, R.J., and Miller, K., "The Moving Finite Element Method: Implementation of a 2-D Code," Presented at the LASL Adaptive Mesh Workshop, Los Alamos, NM; August 5-7, 1981.

3. Doss, S.K., "Solution of the Gas Dynamics Equations in 1-D by the Moving Finite Element Method," presented at SIAM Meeting, Cincinnati, OH, October 26-28, 1981.

4. Gelinas, R.J. and Doss, S.K., "Moving Finite Elements in 2-D," presented at 1982 Army Numerical Analysis and Computers Conference, February 3-4, 1982, Vicksburg, MS.

5. Gelinas, R.J. and Doss, S.K., "The Moving Finite Element Method: Strong Shock and Penetration Mechanics Applications," presented at Army Research Office Workshop on Computational Aspects of Penetration Mechanics, April 27-29, 1982, Aberdeen Proving Ground, MD.

6. Gelinas, R.J., Doss, S.K., Vajk, J.P., Djomehri, M.J., and Miller, K., "Moving Finite Elements in 2-D," prepared for presentation at 10th IMACS Congress, August 8-13, 1982, Montreal, Canada; to appear as full-length article in the Proceedings.

7. Gelinas, R.J. and Doss, S.K., "The Moving Finite Element Method: One-Dimensional Transient Flow Applications," prepared for presentation at 10th IMACS Congress, August 8-13, 1982, Montreal, Canada; to appear in Proceedings.

(ii) Seminars and invited talks on the moving finite element delivered during the past year:

1. Air Force Weapons Laboratory (contact: Dr. Ralph Rudder)
2. Eglin Air Force Base (contact: Major Guy Spitale)
3. Lawrence Livermore Laboratory (contact: Dr. David Anderson)
4. NASA Ames Laboratory (contact: Dr. Paul Cutler)
5. Ford Scientific Laboratory (contact: Dr. A. Paluzny)
6. Sandia Laboratory (July, 1982, contact: Dr. R. Kee)

(iii) Consultive functions to other agencies

1. Eglin Air Force Base; we are currently under contract to investigate the application of the MFE method in 2-D to armor penetration effects; contact: Major Guy Spitale.

2. Defense Nuclear Agency; in negotiation to investigate the application of the MFE method in 2-D to airblast effects; contact: Dr. George Ullrich.

REFERENCES

1. K. Miller and R. Miller, "Moving Finite Elements, Part I" and "Moving Finite Elements, Part II," SIAM J. of Num. Anal., 1019-57, Vol. 18, No. 6, December 1981.

2. R. J. Gelinas, S. K. Doss and K. Miller, "The Moving Finite Element Method: Applications to General Partial Differential Equations with Multiple Large Gradients," J. Comp. Phys., 40, No. 1, p. 202, 1981.

3. R. J. Gelinas and S. K. Doss, "The Moving Finite Element Method: A Semi-Automatic Solver for Diverse PDE Applications," *presented at the Fourth IMACS International Symposium on Computer Methods for Partial Differential Equations, Lehigh University, Bethlehem, PA, June 30-July 2, 1981.*

4. D. Prosnitz, R. A. Haas and S. Doss, "Two-Dimensional Numerical Model of a Free Electron Laser Amplifier," to appear in the J. of Quantum Electronics. (*Summary presented at the Conference on Lasers and Electro-Optics (CLEO '81), Washington, DC, June 10-12, 1981.*)

5. R. J. Gelinas and S. K. Doss, "The Moving Finite Element Method: 1-D Transient Flow Aplications," to be presented at the 10th IMACS World Congress on Systems Simulation and Scientific Computation, Montreal, Canada, August 8-13, 1982.

6. R. J. Gelinas and S. K. Doss, "DYLA - Moving Finite Element Code in 1-D: User Instruction Manual," report for EG&G, Idaho Falls, ID, May 1981. (K. Miller has also developed gradient dependent regularization formulations, unpublished.)

7. R.J., Gelinas, S.K. Doss, J.P. Vajk, J. Djomehri and K. Miller, "Moving Finite Elements in 2-D," to be presented at the 10th IMACS World Congress on Systems Simulation and Scientific Computation, Montreal, Canada, August 8-13, 1982.


ATE
LME